

# TRABALHO RPVMM DESAFIO LOJAS RENNER

UNIVERSIDADE FEDERAL DE SÃO PAULO

# CONTEÚDO

**01**

O PROBLEMA

**02**

A ABORDAGEM

**03**

RESULTADOS E DISCUSSÕES

**04**

CONCLUSÃO

# PROBLEMA

Trata-se de um Order batching por adensamento de estoque.

Contexto do problema:

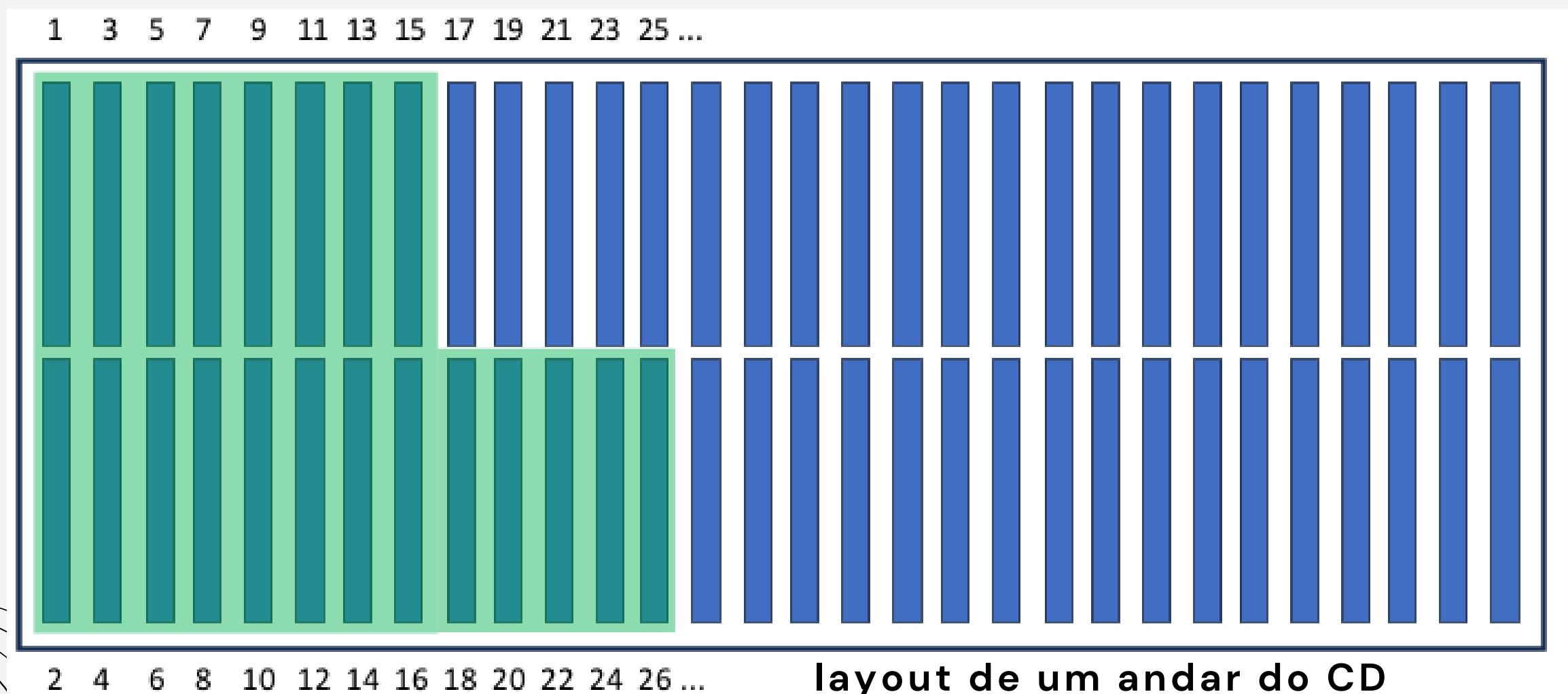
- O Centro de Distribuição (CD) precisa separar produtos para envio às lojas;
- Os produtos são agrupados em caixas planejadas previamente;
- Cada caixa contém diferentes produtos em diferentes quantidades.



# PROBLEMA

## Desafios:

- Produtos podem estar distribuídos em vários corredores.
- O CD possui andares e corredores, tornando a separação complexa.
- Cada onda de produção tem um limite de 6000 peças.
- Só podemos agrupar caixas da mesma classe em uma onda.



## Objetivo:

- Atribuir caixas as ondas;
- Minimizar área de picking! (em verde)

# ABORDAGEM



Optamos por uma abordagem híbrida, mesclando:

- Modelagem matemática;
- Heurística.



Por que essa abordagem?

Apenas modelagem matemática não estava sendo suficiente!



A modelagem oferece maior base sólida.

A abordagem heurística possibilita maior flexibilidade e adaptabilidade ao cenário.



# ABORDAGEM

Inicialmente, pensamos na seguinte abordagem:

- Criar modelagem matemática;
- Aplicar a modelagem em python, com a biblioteca PuLP;
- Minimizar a área de picking obtida pela função objetivo definida.

Para tanto, utilizamos a seguinte modelagem matemática:

Nossas variáveis de decisão são 3:

- $x_{ij}$  : variável que indica se uma caixa 'i' está na onda 'j'.
- $y_{jk}$  : variável que indica se o corredor 'k' é usado pela onda 'j'.
- $z_{jl}$  : variável que indica se um andar 'L' faz parte da onda 'j'.

# ABORDAGEM

É preciso minimizar a seguinte função objetivo:

$$\sum_j \left( \sum_l \left[ \left( \frac{\max A_{jl} - \min A_{jl}}{2} + 1 \right) + \left( \frac{\max B_{jl} - \min B_{jl}}{2} + 1 \right) \right] + \sum_l p \cdot z_{jl} \right)$$

Essa função é um somatório do custo de cada onda. O custo de cada onda é a distância percorrida por corredores junto com uma penalidade para cada andar percorrido.

E as restrições consideradas são:

- Cada caixa 'i' só pode estar associada à uma onda;
- Cada onda só pode ter, no máximo, 6000 peças;
- Idealmente, ondas distintas não **devem** usar os mesmos corredores.

# ABORDAGEM

- Durante a implementação da modelagem com o PuLP, visando uma otimização, não estávamos obtendo resultados satisfatórios.
- O grupo optou por fazer uma abordagem mais voltada para heurística.
- Essa nova abordagem parte de um princípio guloso, levando em consideração a configuração do CD para tomar ações.
- As penalidades apenas influenciaram na área obtida.

# ABORDAGEM

- Após feedback obtido nas apresentações parciais, decidimos fazer uma análise da organização do Centro de Distribuição, a fim de saber qual a melhor abordagem gulosa:
  - CD caótico (desorganizado): priorizar corredores mais densos para criação de ondas.
  - CD organizado: priorizar produtos na criação de ondas, já que eles estão próximos.
- Após análises, percebemos que o dataset fornecido representava um CD organizado.

# ABORDAGEM

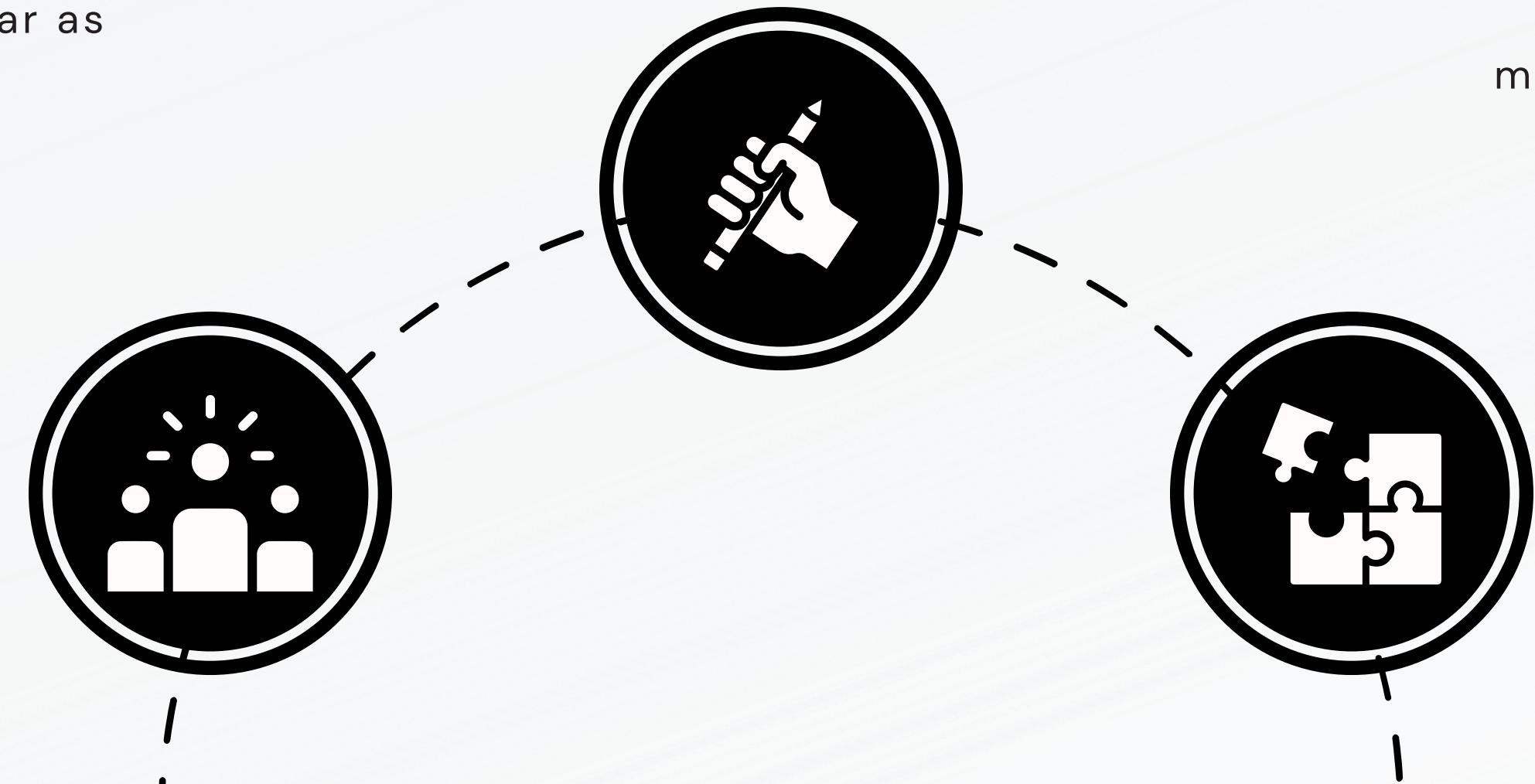
Resumo da abordagem heurística:

## Ponto 1

A heurística inicial pensada foi: partir de corredores mais densos para formar as ondas.

## Ponto 2

Com feedback recebido, optamos por fazer uma análise da organização do CD antes de montar uma heurística.



## Ponto 3

O código agora analisa se um dataset é organizado ou não, escolhendo qual o melhor critério para o algoritmo guloso.

# RESULTADOS

Para a realização dos testes, utilizamos uma base de dados com o seguinte formato:

- (Andar, Corredor, SKU, Peças)

Em que é identificado em qual corredor e andar está um determinado número de peças de determinado código (SKU).

ANDAR	CORREDOR	SKU	PEÇAS
0	2	SKU_17028	193
0	2	SKU_17090	524
0	2	SKU_17035	456
0	2	SKU_18137	324
0	2	SKU_17111	54
0	2	SKU_672	1
0	2	SKU_19519	53
0	2	SKU_2650	10
0	3	SKU_17052	19
0	3	SKU_17111	295
0	3	SKU_19373	92
0	3	SKU_3023	40
0	3	SKU_3024	173
0	3	SKU_743	11
0	3	SKU_2754	30
0	3	SKU_3310	11

Pequena amostra da entrada.

# RESULTADOS

## Priorizando corredores

Essa heurística, para a configuração do CD, não obteve resultados bons.

- Número de ondas geradas: 179
- Área de picking média: 21

Em um CD desorganizado, ela pode vir a ser uma melhor opção.

## Priorizando peças

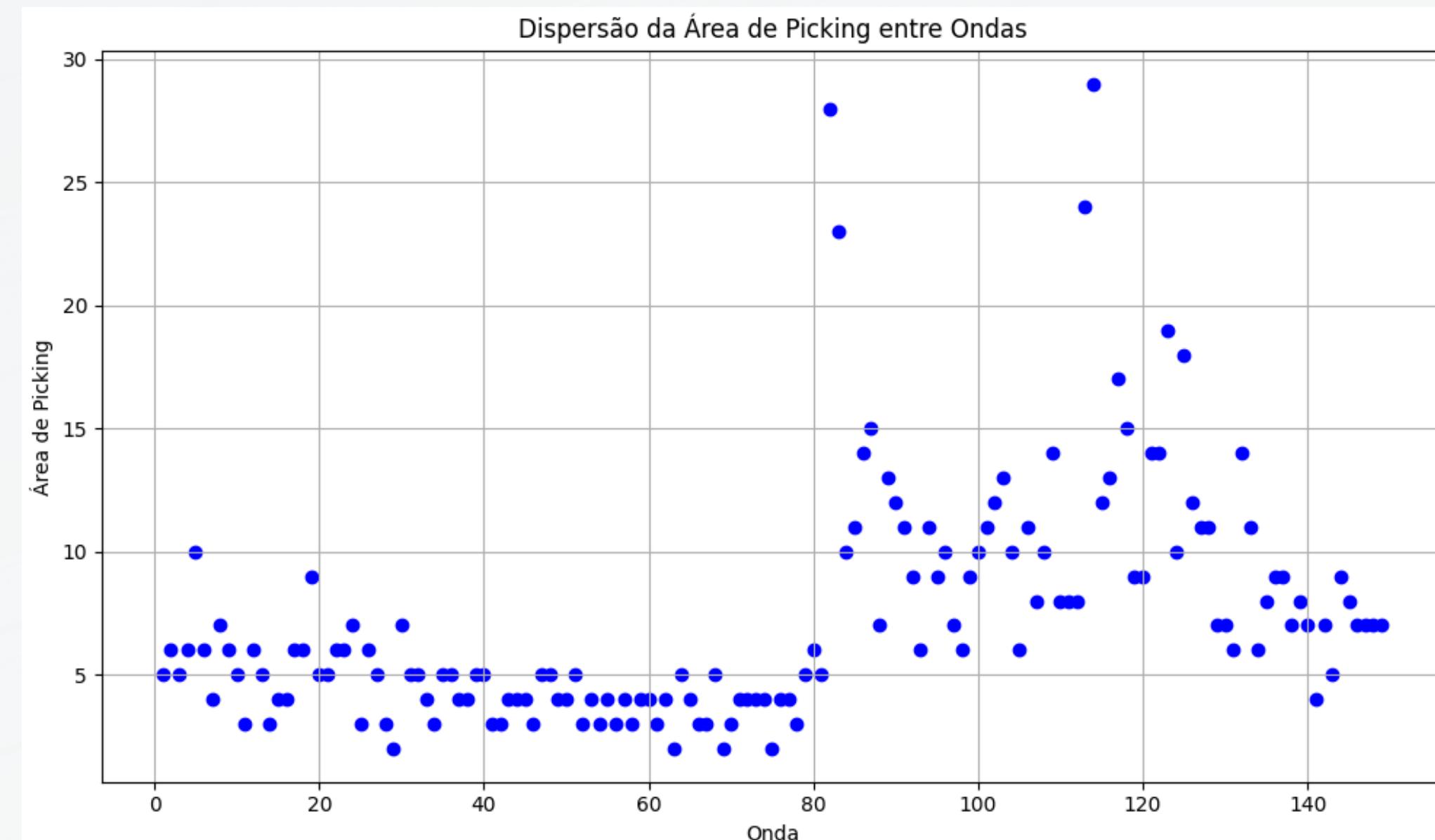
Nesta heurística, o resultado foi mais satisfatório, já que o CD está organizado.

- Número de ondas geradas: 149
- Área de picking média: 8

Para a configuração dada, o resultado obtido foi bom.

# RESULTADOS

## PRIORIZANDO PEÇAS



# CONCLUSÃO

Resumindo, produzimos um código:

- Com abordagem gulosa, focado em produtos;
- Que aproveita o layout passado.

Com a abordagem final, conseguimos:

- Gerar código simples, porém eficiente;
- Execução rápida do algoritmo;
- Obter resultados satisfatórios.

# CONCLUSÃO

O que aprendemos?

- Uma abordagem simples pode ser eficiente quando o ambiente é organizado/controlado;
- Modelagem matemática dá boa noção na busca por soluções.

Melhorias:

- Procurar uma solução mais próxima da otimalidade;
- Testar outras heurísticas;
- Adaptar o código para agir perante configuração do CD;
- Dedicar mais esforços em uma solução de optimização.





# ALGUMA DÚVIDA?



# OBRIGADO!

